

Developing RESTful Services

Famous Ships

Hackathon #2 Requirements

Evaluation Criteria

- Does the web service provide the required service methods? Do the methods operate correctly and return the correct data?
- Are Spring and MyBatis used appropriately and correctly configured? Is auto-wiring used appropriately? Was MyBatis used to develop the interaction with the database? Was the correct data retrieved from the database?
- Have the web service methods been tested using a test Spring context? Does the test class use a Spring context? Are the web service methods adequately tested? Do the service method tests pass?
- Were RESTful standards followed in developing the web service? Do the service methods handle errors properly? Are proper HTTP status codes returned properly?
- Have best practices been followed in all layers? E.g., in Java, Spring, MyBatis, and SQL.

Task

You have recently joined a development team that has been given the task of creating a RESTful service that will provide information about all the famous ships in its database. Your API will provide one web method. The method returns a full list of the ships including their id, name, nickname, captain, ship description, and type.

Your service will return the ship's information in JSON format. In addition, your web service will set appropriate HTTP response statuses:

- If information is present in the database and completes successfully, return status 200
- If the information requested is not present in the database, return HTTP status 400
- If there is any other error during processing, return HTTP status 500

Requirements

- Your service will be implemented with Spring Boot.
- You will use MyBatis for all database interactions.
- The starter project includes a SQL file to create and populate the appropriate tables in Oracle.
- Verify the functionality of your web service methods by writing JUnit tests.
- Use a Spring test context in the tests as appropriate. Ensure that both current and future tests will rollback any changes to the database.
 - Create an integration test for the DAO, MyBatis mapper, and the Oracle database
 - Create unit tests for the POJO
 - Create unit tests for the web layer
 - Create an E2E test